Will USB have an impact on test-and-measurement users?

Edited by Senior Tech Editor Alex Mendelsohn

eeProductCenter
(**04/05/2005** 11:09 AM ET)

There are quite a few incremental changes happening in the test industry, especially with respect to communications buses. Nonetheless, the communication bus isn't as important as how you communicate with your instruments.

By Eric Singer, Senior Software Engineer, National Instruments, Austin, Texas

In late 2002, the *USBTMC* (Universal Serial Bus Test & Measurement Class) specification was released. As USBTMC instruments become available, many may wonder how this new protocol will affect the test-and-measurement industry.

Manufacturers now have a choice of including USB on their instruments, Likewise, test engineers can now use USB as a framework to create automated test systems, along with *IEEE-488/GPIB, PCI, PXI*, and *Ethernet*.

Table – Instrument Control Bus Comparison

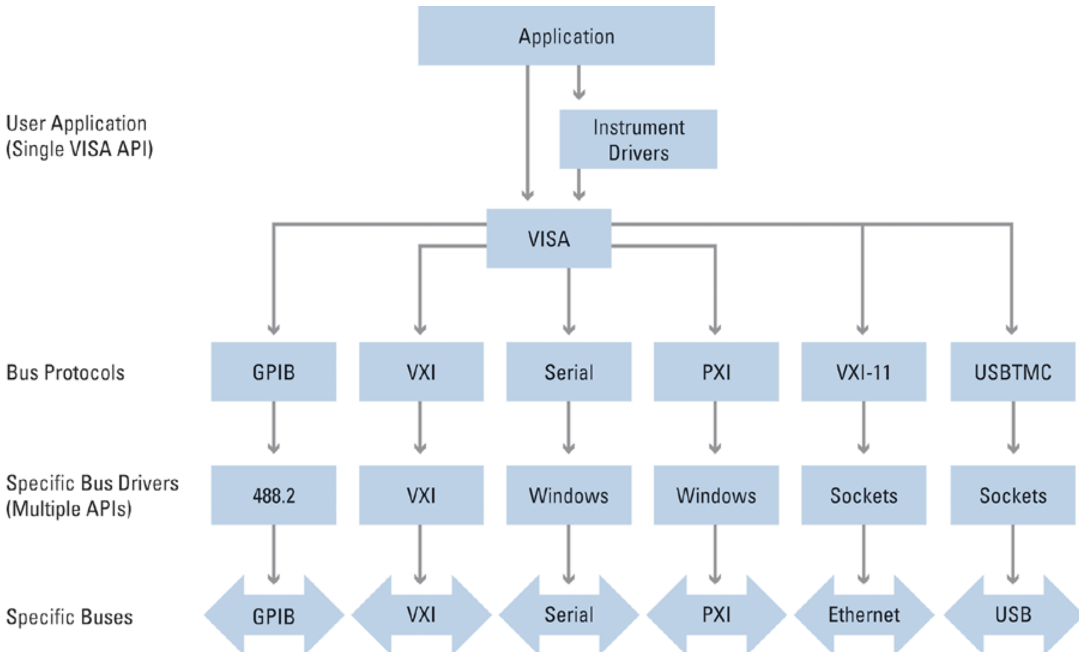
	GPIB	**RS-232**	**Ethernet**	**USB**	**IEEE 1394**
Max. Throughput	1.5 – 2 MB/s 8 MB/s (HS488)	28.8 kB/s	125 MB/s	60 MB/s	100 MB/s
Product Availability	>10,000	>10,000	<100	<100	<100
Latency (1st Byte, etc.)	Low	Low/Medium	High	High	High
Max. # of Devices	14	1	No Limit	127	63
Max. Cable Length	20 m	15 m	No Limit (300 ft Wireless)	30 m	4.5 m
Connector Type	Industrial	Industrial/ Consumer	Consumer	Consumer	Consumer
Triggering Support	Minimal	Minimal	Minimal	Minimal	Minimal

But, is USBTMC going to shake up the entire T&M industry and change how things are done?

The Software is the Key

Twenty years ago, when GPIB was the only available test-and-measurement bus, most people wrote test applications using GPIB-specific calls. When a new bus was introduced, engineers had to learn not only its idiosyncrasies, but also new communication protocols and programming methodologies.

Then, *National Instruments*, along with *GenRad, Racal Instruments*, *Tektronix*, and *Wavetek*, founded the *VXIplug&play Systems Alliance* in 1993. In 1995, the VXIplug&play Systems Alliance defined *VISA*, the Virtual Instrument Software Architecture. It gave test engineers a new way to look at instrument control.



VISA changed instrument control and the test industry by abstracting the differences between buses, incorporating bus protocols, and providing a unified *API* (application programming interface) to communicate to test equipment—regardless of communication bus.

No More Complicated Strings

By incorporating VISA technology, instrument drivers—which focus on instrument functionality instead of instrument communication—became the way to communicate to test equipment. Test applications were no longer forced to construct the complicated strings the instrument accepts to configure it.

Instead, applications could have the instrument driver configure the instrument and create appropriate strings. Because VISA technology is used in instrument drivers, the same instrument driver works with an instrument, regardless of the communication bus.

Because of these benefits, VISA is now *the* T&M standard for instrument communication. It's supported by all major instrument vendors. It also includes support for all major instrumentation buses (such as GPIB, serial, Ethernet, USB, and VXI). In addition, the National Instruments implementation of VISA includes support for PXI instruments.

Instrument drivers that conform to the VXIplug&play or *IVI* (Interchangeable Virtual Instruments) instrument driver standards are generally required to use VISA as the underlying I/O transport. In 1996, National Instruments created the instrument driver network . The instrument driver network lets test engineers find the driver they need online, download it, and then immediately incorporate it into their applications. There are instrument drivers for over 4000 instruments available on-line.

Getting The Right Instrument

Because of VISA and instrument drivers, test engineers don't need to worry about which bus they're using when they write a program to connect their instruments. But there is a catch. Engineers need to be using the right instrument for this to work.

Since a standard VISA instrument session (also known as an *INSTR* session) communicates using industry standard protocols, to communicate with an Ethernet instrument, you need to use an instrument that complies with the *TCP/IP Instrument Protocol Specification* (also known as *VXI-11*).

Some instrument manufacturers use proprietary Ethernet protocols to communicate to their instrument. Be wary of these instruments, as it will be a lot harder to re-use your communication code or replace the instrument if necessary.

The same concern applies to USB instruments. VISA supports the industry-standard *USBTMC* and *USBTMC-USB488* protocols. A USB instrument that isn't one of these may be programmed differently than its GPIB counterpart. The best instruments are ones that have multiple bus options, and follow industry-standard protocols supported by VISA.

Multiple-Bus Synergy

GPIB, Ethernet, and USB are all viable bus options, and each one has its pros and cons. The difficult task for a test engineer is determining which bus is the best one for a given application.

For example, GPIB has been around for over 30 years; today over 90% of T&M instruments offer a GPIB communication option.

This is because GPIB is still an excellent choice for most instrument control applications. It has very low first-byte latencies, so instrument-control applications that contain mostly small transfers tend to be faster across GPIB than with any other bus. It also has robust connectors, well-shielded cables, and a protocol that is completely handled in hardware, permitting a high-quality GPIB instrument to be easily created.

However, GPIB cables can be big and bulky, and overall throughput tops out at 1.5-Mbytes/s (or, 12-Mbits/s), so large data transfers may be less efficient than with other buses. Instruments that support the IEEE-488.1-2003 non-interlocked handshake can increase this rate to 8-Mbytes/s (64-Mbits/s) to make large transfers more efficient.

In contrast, USB and Ethernet have large first-byte latencies, making small transfers less efficient. USB, however, with its bandwidth of 480-Mbits/s (actually about 400-Mbits/s after factoring-in protocol and bus overhead), is very efficient at moving large blocks of data.

In addition, the availability of USB makes it attractive to desktop applications where instrument portability is very important. National Instruments has a treatise entitled *The Future for New Bus Technologies in Instrument Control and Connectivity*. Available on the company's Web site, it discusses multi-bus synergy.

Multiple-Bus Test Systems

A test system contains one or more instruments that communicate using a bus such as GPIB, VXI, PXI, Ethernet, or serial. As more T&M buses become available, single-bus test systems will be replaced by multiple-bus test systems.

The advantage of a multiple-bus test system is that it lets test engineers connect an instrument to the bus that best matches with its characteristics. You can, for example, use GPIB instruments to do most of the work, Ethernet instruments to share among test systems, and USB instruments to handle large data sets.

In addition to these common message-based instrument buses, you can incorporate register-based instrument buses such as VXI and PXI. The register-based buses don't communicate with strings, so there is no overhead of parsing.

In addition, these register-based buses have low latencies and high bandwidth. For example, PXI has latencies in the sub-microsecond level and a bandwidth of over 1-Gbit/s.

Fortunately all of these instruments can be controlled with instrument drivers that communicate using VISA, with no worrying about the underlying protocol.

So, let's go back to the original question: is USBTMC going to shake up the entire T&M industry and change how things are done?

The answer is: not really.

Each bus has its merits and which one to use should be based on your application. With the availability of USBTMC-USB488 instruments, test engineers now have another choice in instrument connectivity, letting them make small modifications to their test setup to take advantage of the unique characteristics of USB.

However, this is just a small incremental advantage over the test systems that were created years ago. The communication bus isn't as important as how you communicate to your instrument!

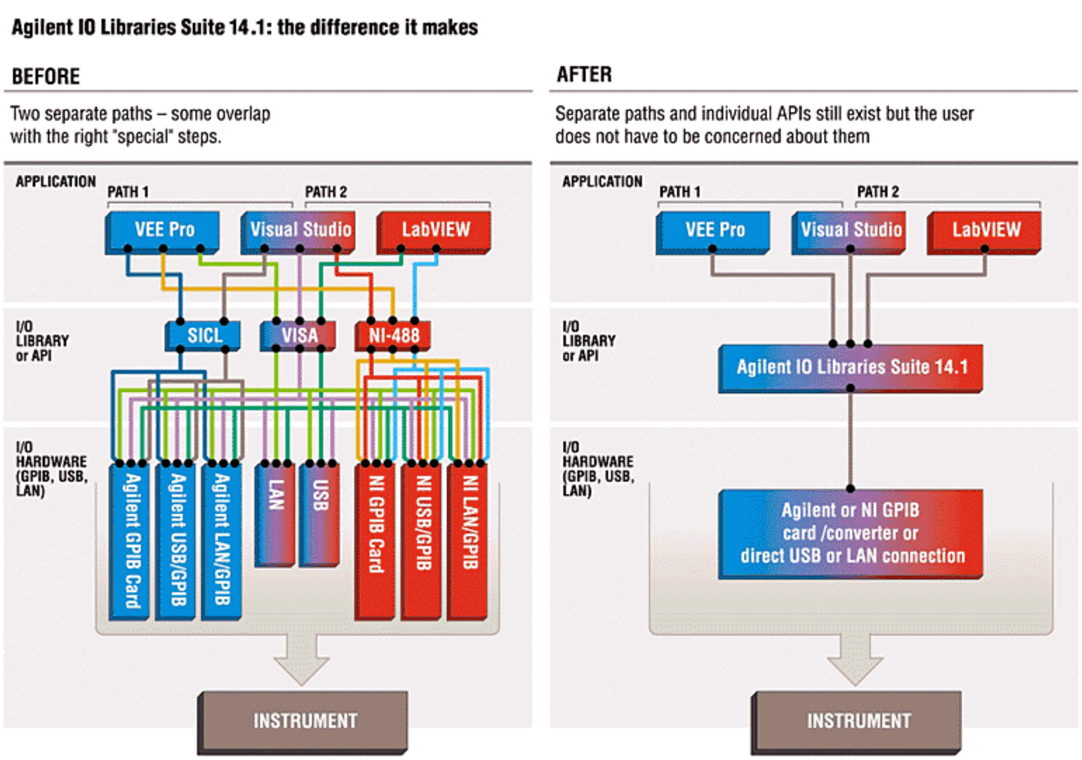
Due to the flexibility they offer, test engineers should look for instruments that support multiple communication buses. In addition, using instruments that communicate with industry-standard protocols and have a VISA-based instrument driver gives you an advantage, as your code will work the same regardless of the bus you choose.

Click here to read a brief tutorial on USBTMC and USBTMC-USB488.

About the Author

Author Eric Singer is a Senior Software Engineer in Instrument Control. He represents National Instruments within the USB Test and Measurement committee where he helped develop USBTMC and USBTMC-USB488. Eric is a graduate of Texas A&M University.

http://www.home.agilent.com/upload/cmc_upload/All/IOLib14.1Diag800x555_5.gif



IO Libraries suite 14.1 esimated ship date Aug 15, 2005